

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO MÔN HỌC
PHÁT TRIỂN TRÒ CHƠI

ĐỒ ÁN CUỐI KỲ

Người hướng dẫn: **GV. VŨ ĐÌNH HỒNG**

Người thực hiện: **NGUYỄN MINH GIANG – 51800767**

CAO THANH PHONG – 51800465

TRƯƠNG MINH HẬU – 51800383

THẠCH LƯU HOÀNG ANH - 51800345

Nhóm : 1

Khoá : 22

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020

LỜI CẢM ƠN

Để hoàn thành báo cáo này, lời đầu tiên nhóm em xin gửi lời cảm ơn đến các tổ chức, nhà trường đã tạo điều kiện giúp đỡ.

Thứ hai, nhóm em xin gửi lời cảm ơn đến thầy Vũ Đình Hồng - giảng viên bộ môn Phát triển trò chơi đã cung cấp cho chúng em những kiến thức hữu ích để hoàn thành bài báo cáo này.

Cuối cùng, nhóm em xin chân thành cảm ơn gia đình và bạn bè, đã luôn tạo điều kiện, quan tâm, giúp đỡ, động viên trong suốt quá trình học tập và hoàn thành bài tiểu luận này.

BÁO CÁO ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là báo cáo của riêng tôi và được sự hướng dẫn của GV Vũ Đình Hồng. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong báo cáo còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung báo cáo của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm
2020*

Tác giả

(ký tên và ghi rõ họ tên)

MỤC LỤC

LỜI CẢM ƠN.....	2
BÁO CÁO ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG	3
MỤC LỤC.....	4
MỤC LỤC HÌNH ẢNH VÀ BẢNG BIỂU	6
MỤC LỤC HÌNH ẢNH	6
MỤC LỤC BẢNG BIỂU	6
CHƯƠNG I – GIỚI THIỆU ĐỀ TÀI.....	7
I. Sơ lược đề tài:	7
4. Tác giả:.....	7
5. Cốt truyện:.....	7
6. Luật chơi:.....	7
8. Feature Set:.....	8
II. Lý do chọn đề tài:	8
III. Yêu cầu đề tài:	8
1. Map:.....	8
2. Nhân vật:.....	8
3. Âm thanh:	8
4. Camera:	8
IV. Cơ sở lý thuyết:	9
CHƯƠNG II – PHÂN TÍCH THIẾT KẾ.....	10
I. Xác định Actor và Usecase	10
1. Actor:	10
2. Usecase:.....	10
II. Sơ đồ Usecase tổng quát:	10
III. Đặc tả Usecase	11
1. UC Di chuyển.....	11
2. UC Tấn công Monster	15
3. UC Di chuyển tự động	20
4. UC Tấn công Player.....	21
CHƯƠNG III – HIỆN THỰC HỆ THỐNG.....	23

CHƯƠNG IV – KẾT QUẢ ĐẠT ĐƯỢC	33
CHƯƠNG V – KẾT LUẬN	34
I. Ưu điểm:	34
II. Nhược điểm:	34
III. Những gì đã làm được:	34
IV. Hướng phát triển cho tương lai:	34
TỔNG KẾT	35
TÀI LIỆU THAM KHẢO	36

MỤC LỤC HÌNH ẢNH VÀ BẢNG BIỂU

MỤC LỤC HÌNH ẢNH

Hình 1 - Sơ đồ Usecase tổng quát.....	11
Hình 2 - UC Di chuyển	11
Hình 3 - UC Tấn công Monster	15
Hình 4 - UC Di chuyển tự động.....	20
Hình 5 - UC Tấn công Player	21
Hình 6 - Map Level 1-1	23
Hình 7 - Map Level 1-2	24
Hình 8 - Map Level 2-1	24
Hình 9 - Map level 2-2	25
Hình 10 - Map level 3-1	25
Hình 11 – Menu đầu tiên.....	26
Hình 12 - Menu End game.....	26
Hình 13 - Menu Game Over.....	27
Hình 14 - Logic animation Player	27
Hình 15 – Màn hình vào game.....	33

MỤC LỤC BẢNG BIỂU

Hình 1 - Sơ đồ Usecase tổng quát.....	11
Hình 2 - UC Di chuyển	11
Hình 3 - UC Tấn công Monster	15
Hình 4 - UC Di chuyển tự động.....	20
Hình 5 - UC Tấn công Player	21
Hình 6 - Map Level 1-1	23
Hình 7 - Map Level 1-2	24
Hình 8 - Map Level 2-1	24
Hình 9 - Map level 2-2	25
Hình 10 - Map level 3-1	25
Hình 11 – Menu đầu tiên.....	26
Hình 12 - Menu End game.....	26
Hình 13 - Menu Game Over.....	27
Hình 14 - Logic animation Player	27
Hình 15 – Màn hình vào game.....	33

CHƯƠNG I – GIỚI THIỆU ĐỀ TÀI

I. Sơ lược đề tài:

1. **Đề tài:** Game 2D offline thể loại có cốt truyện

2. **Tên game:** Ninja cứu mẹ

3. **Version:** V0.1

4. **Tác giả:**

- Cao Thanh Phong – 51800465
- Trương Minh Hậu – 51800383
- Thạch Lưu Hoàng Anh – 51800345
- Nguyễn Minh Giang – 51800767

5. **Cốt truyện:**

Tại vùng đất Nhật Bản xa xưa, Magun cùng mẹ sống trong ngôi nhà nhỏ ở chân núi. Anh là một ninja tài năng với nhiều bí thuật anh học được trong cuốn trục mà người cha đã mất để lại. Magun rất chăm chỉ học để trau dồi kỹ năng, đồng thời sử dụng kỹ năng đã học thực hiện những nhiệm vụ khó khăn nhằm trang trải cuộc sống của hai mẹ con. Trong khi đó, ở phía Nam nước Nhật, một đám quái vật do S, hay còn gọi là Boss, cầm đầu đang nhắm nhe cuốn trục bí thuật với ham muốn chinh phục thế giới. Bọn chúng đã lập ra kế hoạch bắt cóc mẹ của Magun và ép Magun phải giao ra cuốn trục. Magun, sau khi mẹ bị bắt, đã nhanh chóng lên đường, vượt qua thử thách cùng sự trợ giúp của những người dân tốt bụng trên đường đi, quyết tâm đánh bại quái vật để cứu mẹ và cứu cả thế giới.

Đến với “Ninja cứu mẹ”, người chơi sẽ nhập vai Magun và trải nghiệm cuộc hành trình cứu mẹ đầy gian nan nhưng cũng vô cùng ý nghĩa. Người chơi sẽ tận tay tiêu diệt quái vật, vượt qua thử thách để đến được căn cứ chính, chiến đấu và cứu người mẹ. Trò chơi kết thúc khi Magun hạ gục Boss, cứu mẹ thành công.

6. **Luật chơi:**

- Người chơi nhập vai Ninja di chuyển và thao tác từ bàn phím:
 - o Phím mũi tên: di chuyển
 - o Phím A: đánh thường
 - o Phím S: phun lửa (số lượng lửa giới hạn)
 - o Phím D: tuyệt chiêu (có thời gian hồi chiêu)
 - o Phím Shift: trượt
- Ninja vượt vòng theo thứ tự
- Ninja có 3 mạng, sau khi mất hết 3 mạng sẽ quay lại điểm xuất phát

- Máu của ninja sẽ mất dần nếu chịu sự tấn công của quái vật và ngược lại
- Tùy vào mỗi loại quái có thể có cơ chế hồi sinh/ chuyển dạng.
- Hạ gục Boss để thắng trò chơi.

7. Độ tuổi giới hạn: 6+

8. Feature Set:

- Levels: 3 levels
- Đồ họa: 2D
- Ngôn ngữ lập trình: C#
- Công cụ: Unity

II. Lý do chọn đề tài:

Đây là một đề tài hấp dẫn, mang đến cho người chơi cả những cảm giác cũ và mới. Người chơi vừa có thể trải nghiệm lại phong cách chơi cổ điển, vừa có thể trải nghiệm những tính năng mới hơn với “Ninja cứu mẹ”.

Đề tài sở hữu đồ họa 2D cổ điển, ít dung lượng sẽ tạo ra dấu ấn giữa rừng game 3D đang phổ biến hiện nay. Hơn nữa, đề tài game offline có cốt truyện mang đến cho người chơi cảm giác được hòa mình vào câu chuyện cứu mẹ đầy cảm động của Magun và có thể chơi mọi lúc dù không có wifi hay dữ liệu.

Đó là những lý do chính để nhóm chọn đề tài game 2D offline có cốt truyện.

III. Yêu cầu đề tài:

1. Map:

- Đồ họa: 2D
- Nhiều địa hình

2. Nhân vật:

- Đồ họa: 2D
- Có thể di chuyển
- Có thể tung chiêu thức để hạ gục đối phương
- Đụng phải quái sẽ mất mạng (trái tim)
- Chết khi hết mạng
- Sau khi chết chuyển tới giao diện đầu.

3. Âm thanh:

- Âm thanh nền phù hợp, dễ nghe
- Có âm thanh khi tung chiêu

4. Camera:

- Camera di chuyển theo nhân vật chính (Ninja)

IV. Cơ sở lý thuyết:

- Áp dụng lý thuyết về Unity
- Lý thuyết học trên lớp
- Slide bài giảng
- Các bài học trên lớp thực hành
- Video hướng dẫn
- Kiến thức từ giáo viên

CHƯƠNG II – PHÂN TÍCH THIẾT KẾ

I. Xác định Actor và Usecase

1. Actor:

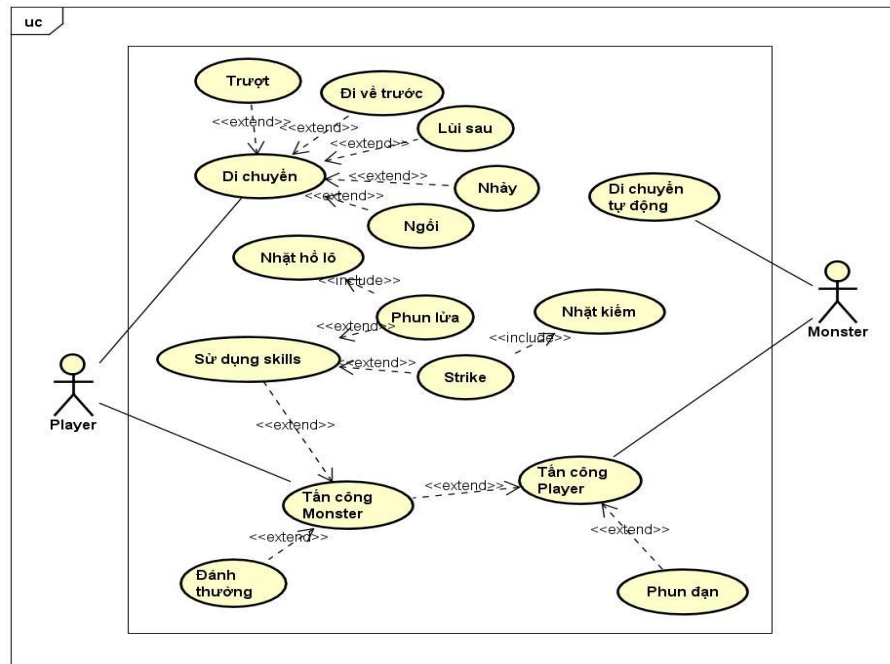
- Player: Ninja do người chơi nhập vai, có thể sử dụng các kỹ năng được lập trình sẵn.
- Monster: Quái vật trong game, sử dụng các chiêu thức được lập trình để hạ gục người chơi.

2. Usecase:

ID	Usecase	Mô tả	Actor
UC01	Di chuyển	Người chơi nhập vai Ninja và thao tác các di chuyển bằng bàn phím.	Player
UC02	Tấn công Monster	Người chơi nhập vai Ninja sử dụng các skills hoặc đánh thường để hạ quái vật.	Player
UC03	Di chuyển tự động	Quái vật hệ thống được lập trình để di chuyển tự động, có thể đuổi theo người chơi.	Monster
UC04	Tấn công Player	Quái vật hệ thống có thể tung chiêu để hạ gục người chơi.	Monster

Bảng 1 - Mô tả Usecase chính

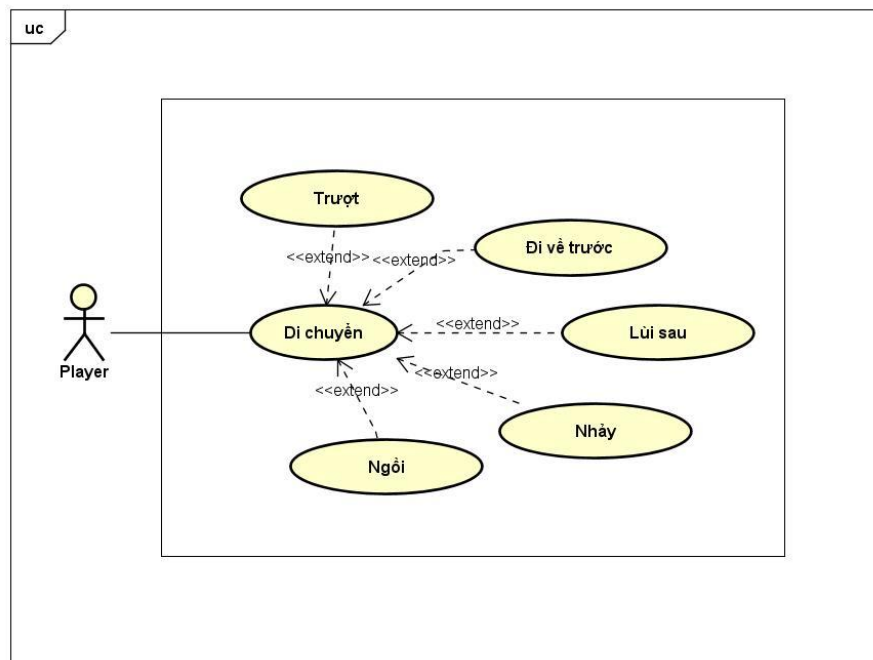
II. Sơ đồ Usecase tổng quát:



Hình 1 - Sơ đồ Usecase tổng quát

III. Đặc tả Usecase

1. UC Di chuyển



Hình 2 - UC Di chuyển

1.1. UC Di chuyển

ID	UC01
----	------

Tên usecase	Di chuyển	
Mô tả	Người chơi nhập vai Ninja và thao tác các di chuyển bằng bàn phím.	
Actor	Player	
Usecase liên quan	Trượt, Đi về trước, Lùi sau, Nhảy, Ngồi	
Actor liên quan	Không	
Sự kiện kích hoạt	Người chơi nhấn các phím mũi tên để di chuyển	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Sau khi bắt đầu trò chơi, nhấn các phím mũi tên để di chuyển	1.1.Nhân vật di chuyển theo điều khiển của người chơi
Kết quả	Nhân vật di chuyển trong map trò chơi.	

Bảng 2 - UC Di chuyển

1.2. UC Di chuyển – Trượt

ID	UC01.1	
Tên usecase	Trượt	
Mô tả	Người chơi nhập vai Ninja và muốn nhân vật trượt trong map.	
Actor	Player	
Usecase liên quan	Di chuyển	
Actor liên quan	Không	
Sự kiện kích hoạt	Người chơi nhấn Shift + các phím mũi tên (◀ hoặc ▶) để trượt	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Sau khi bắt đầu trò chơi, nhấn Shift + các phím mũi tên (◀ hoặc ▶) để trượt	1.1.Nhân vật trượt theo hướng điều khiển của người chơi
Kết quả	Nhân vật trượt trong map trò chơi theo hướng điều khiển.	

Bảng 3 - UC Trượt

1.3. UC Di chuyển – Đi về trước

ID	UC01.2	
Tên usecase	Đi về trước	
Mô tả	Người chơi nhập vai Ninja và muốn nhân vật đi về trước trong map.	
Actor	Player	
Usecase liên quan	Di chuyển	
Actor liên quan	Không	
Sự kiện kích hoạt	Người chơi nhấn phím mũi tên (▶) để di chuyển về trước	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Sau khi bắt đầu trò chơi, nhấn phím mũi tên (▶) để di chuyển về trước	1.1. Nhân vật đi về trước theo điều khiển của người chơi
Kết quả	Nhân vật đi về trước trong map trò chơi theo điều khiển.	

Bảng 4 - UC Đi về trước

1.4. UC Di chuyển – Lùi sau

ID	UC01.3	
Tên usecase	Lùi sau	
Mô tả	Người chơi nhập vai Ninja và muốn nhân vật lùi sau trong map.	
Actor	Player	
Usecase liên quan	Di chuyển	
Actor liên quan	Không	
Sự kiện kích hoạt	Người chơi nhấn phím mũi tên (◀) để di chuyển lùi về sau	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Sau khi bắt đầu trò chơi, nhấn phím mũi tên (◀) để di chuyển lùi về sau	1.1. Nhân vật lùi về sau theo điều khiển của người chơi
Kết quả	Nhân vật lùi sau trong map trò chơi theo điều khiển.	

Bảng 5 - UC Lùi sau

1.5. UC Di chuyển – Nhảy

ID	UC01.4	
Tên usecase	Nhảy	
Mô tả	Người chơi nhập vai Ninja và muốn nhân vật nhảy trong map.	
Actor	Player	
Usecase liên quan	Di chuyển	
Actor liên quan	Không	
Sự kiện kích hoạt	Người chơi nhấn phím mũi tên (▲) để nhảy	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Sau khi bắt đầu trò chơi, nhấn phím mũi tên (▲) để nhảy	1.1. Nhân vật nhảy theo điều khiển của người chơi
Kết quả	Nhân vật nhảy theo điều khiển.	

Bảng 6 - UC Nhảy

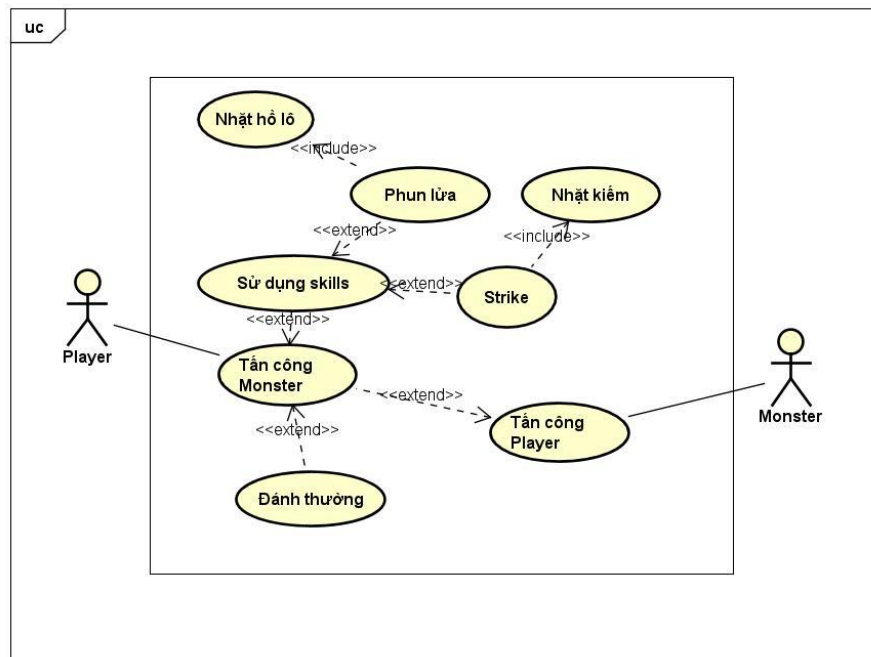
1.6. UC Di chuyển – Ngồi

ID	UC01.5	
Tên usecase	Ngồi	

Mô tả	Người chơi nhập vai Ninja và muốn nhân vật ngồi trong map.	
Actor	Player	
Usecase liên quan	Di chuyển	
Actor liên quan	Không	
Sự kiện kích hoạt	Người chơi nhấn phím mũi tên (▼) để ngồi	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Lưuồng sự kiện	Tác nhân	Hệ thống
	1.Sau khi bắt đầu trò chơi, nhấn phím mũi tên (▼) để ngồi	1.1. Nhân vật ngồi theo điều khiển của người chơi
Kết quả	Nhân vật ngồi theo điều khiển.	

Bảng 7 - UC Ngồi

2. UC Tấn công Monster



Hình 3 - UC Tấn công Monster

2.1. UC Tấn công Monster

ID	UC02
Tên usecase	Tấn công Monster
Mô tả	Người chơi nhập vai Ninja sử dụng các skills hoặc đánh thương để hạ quái vật.
Actor	Player

Usecase liên quan	Sử dụng skills, Đánh thường, Tấn công Player	
Actor liên quan	Monster	
Sự kiện kích hoạt	Người chơi nhấn các phím chữ để tấn công quái vật	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Điều kiện sau	Người chơi gặp quái vật	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Gặp quái vật 2. Nhấn phím chữ để tấn công	2.1.Nếu tấn công trúng, quái vật biến mất, cộng điểm cho người chơi 2.2.Nếu tấn công trượt, quái vật không biến mất
Kết quả	Người chơi tấn công và có thể tiêu diệt quái vật	

Bảng 8 - UC Tấn công Monster

2.2. UC Tấn công Monster - Đánh thường

ID	UC02.1	
Tên usecase	Đánh thường	
Mô tả	Người chơi nhập vai Ninja đánh thường để hạ quái vật.	
Actor	Player	
Usecase liên quan	Tấn công Monster	
Actor liên quan	Monster	
Sự kiện kích hoạt	Người chơi nhấn phím “A” để tấn công quái vật	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Điều kiện sau	Người chơi gặp quái vật	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Gặp quái vật 2. Nhấn “A” để đánh thường	2.1.Nhân vật đánh thường theo tầm đánh được thiết lập 2.2.Nếu tấn công trúng, quái vật biến mất, cộng điểm cho người chơi 2.3.Nếu tấn công trượt, quái vật không biến mất
Kết quả	Người chơi đánh thường và có thể tiêu diệt quái vật	

Bảng 9 - UC Đánh thường

2.3. UC Tấn công Monster – Sử dụng Skills

2.3.1. UC Sử dụng Skills

ID	UC02.2	
Tên usecase	Sử dụng Skills	
Mô tả	Người chơi nhập vai Ninja sử dụng các skills để hạ quái vật.	
Actor	Player	
Usecase liên quan	Phun lửa, Strike	
Actor liên quan	Monster	
Sự kiện kích hoạt	Người chơi nhấn phím “S” hoặc “D” để tấn công quái vật	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Điều kiện sau	Người chơi gặp quái vật	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Gặp quái vật 2. Nhấn “S” hoặc “D” để sử dụng skills	2.1. Nhân vật sử dụng skills được thiết lập sẵn 2.2. Nếu tấn công trúng, quái vật biến mất, cộng điểm cho người chơi 2.3. Nếu tấn công trượt, quái vật không biến mất
Kết quả	Người chơi sử dụng skills và có thể tiêu diệt quái vật	

Bảng 10 - UC Sử dụng Skills

2.3.2. UC Phun lửa

ID	UC02.2.1
Tên usecase	Phun lửa
Mô tả	Người chơi nhập vai Ninja phun lửa để hạ quái vật.
Actor	Player
Usecase liên quan	Nhặt hồ lô, Sử dụng skills
Actor liên quan	Monster
Sự kiện kích hoạt	Người chơi nhấn phím “S” để phun lửa tấn công quái vật
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi
Điều kiện sau	<ul style="list-style-type: none"> - Người chơi gặp quái vật - Người chơi nhặt được hồ lô

Luồng sự kiện	Tác nhân	Hệ thống
	<ol style="list-style-type: none"> 1. Gặp quái vật 2. Nhấn “S” để phun lửa 	<ol style="list-style-type: none"> 2.1. Nhân vật phun lửa theo tầm tấn công được thiết lập và theo số lửa đang có. 2.2. Nếu tấn công trúng, quái vật biến mất, cộng điểm cho người chơi 2.3. Nếu tấn công trượt, quái vật không biến mất
Kết quả	Người chơi phun lửa và có thể tiêu diệt quái vật	
Ngoại lệ	Hết số lửa đang có: Không thể phun ra lửa nữa.	

Bảng 11 - UC Phun lửa

2.3.3. UC Nhặt hồ lô

ID	UC02.2.1.1	
Tên usecase	Nhặt hồ lô	
Mô tả	Người chơi nhập vai Ninja nhặt hồ lô để có thể thực hiện skills phun lửa	
Actor	Player	
Usecase liên quan	Phun lửa	
Sự kiện kích hoạt	Người chơi di chuyển chạm biểu tượng hồ lô	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Điều kiện sau	Người chơi gặp biểu tượng hồ lô	
Luồng sự kiện	Tác nhân	Hệ thống
	<ol style="list-style-type: none"> 1. Sử dụng các phím mũi tên để di chuyển 2. Chạm biểu tượng hồ lô 	<ol style="list-style-type: none"> 2.1. Biểu tượng lửa ở góc trái trên màn hình tăng lên số lượng tương ứng 2.2. Thiết lập nhân vật có thể sử dụng skill phun lửa theo số lửa ở góc trái trên màn hình
Kết quả	Người chơi nhặt được hồ lô và có thể sử dụng skills phun lửa	

Bảng 12 - UC Nhật hồ lô

2.3.4. UC Strike

ID	UC02.2.2	
Tên usecase	Strike	
Mô tả	Người chơi nhập vai Ninja sử dụng tuyệt chiêu để hạ quái vật.	
Actor	Player	
Usecase liên quan	Nhật kiếm, Sử dụng skills	
Actor liên quan	Monster	
Sự kiện kích hoạt	Người chơi nhấn phím “D” để sử dụng tuyệt chiêu tấn công quái vật	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Điều kiện sau	<ul style="list-style-type: none"> - Người chơi gặp quái vật - Người chơi nhật được kiếm 	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Gặp quái vật 2. Nhấn “D” để dùng tuyệt chiêu	2.1. Nhân vật tung ra tuyệt chiêu với sát thương lớn 2.2. Nếu tấn công trúng, quái vật biến mất, cộng điểm cho người chơi 2.3. Nếu tấn công trượt, quái vật không biến mất
Kết quả	Người dùng tuyệt chiêu và có thể tiêu diệt quái vật	
Ngoại lệ	Tuyệt chiêu chưa hồi (hồi trong 10s): Nhân vật không thể dùng tuyệt chiêu	

Bảng 13 - UC Strike

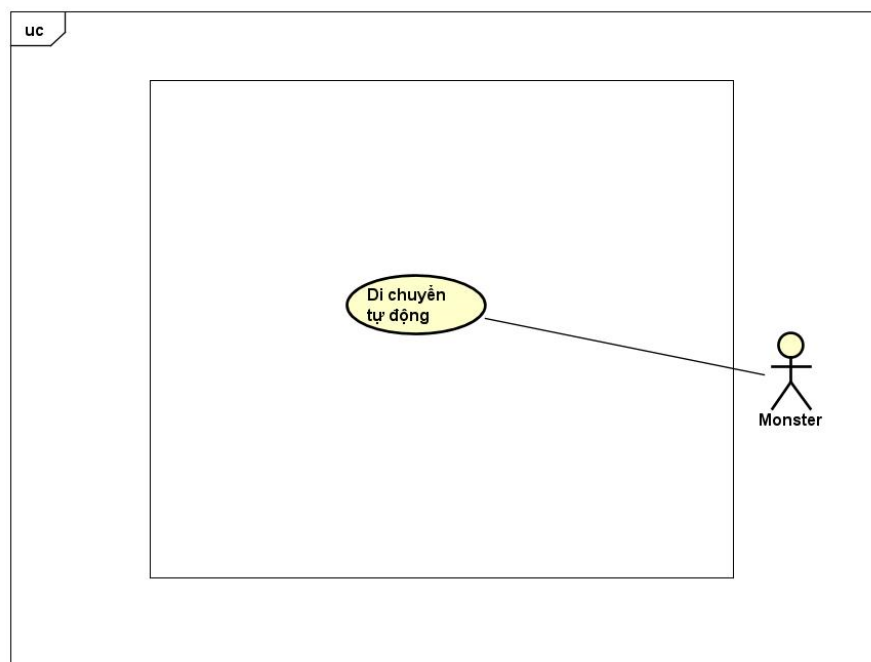
2.3.5. UC Nhật kiếm

ID	UC02.2.2.1
Tên usecase	Nhật kiếm
Mô tả	Người chơi nhập vai Ninja nhật kiếm để có thể thực hiện tuyệt chiêu
Actor	Player
Usecase liên quan	Strike
Sự kiện kích hoạt	Người chơi di chuyển chạm biểu tượng thanh kiếm

Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Điều kiện sau	Người chơi gặp biểu tượng thanh kiếm	
Lưuồng sự kiện	Tác nhân	Hệ thống
	1.Sử dụng các phím mũi tên để di chuyển 2.Chạm biểu tượng thanh kiếm	1.1. Thiết lập nhân vật có thể sử dụng tuyệt chiêu 1.2. Thiết lập thời gian hồi chiêu sau khi sử dụng
Kết quả	Người chơi nhặt được kiếm và có thể sử dụng tuyệt chiêu để tấn công quái vật.	

Bảng 14 - UC Nhặt kiếm

3. UC Di chuyển tự động



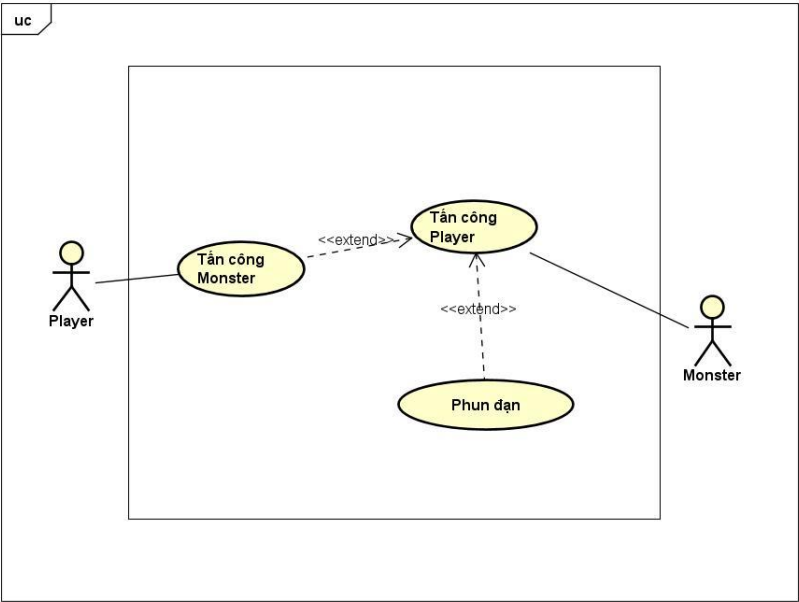
Hình 4 - UC Di chuyển tự động

ID	UC03
Tên usecase	Di chuyển tự động
Mô tả	Quái vật hệ thống được lập trình để di chuyển tự động, có thể đuổi theo người chơi.
Actor	Monster
Sự kiện kích hoạt	Trò chơi tại map có quái vật bắt đầu

Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Điều kiện sau	- Map có quái vật	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Nhấn bắt đầu trò chơi	1.1.Quái vật di chuyển theo thiết lập
Kết quả	Quái vật di chuyển tự động theo thiết lập	

Bảng 15 - UC Di chuyển tự động

4. UC Tấn công Player



Hình 5 - UC Tấn công Player

4.1. UC Tấn công Player

ID	UC04	
Tên usecase	Tấn công Player	
Mô tả	Quái vật hệ thống có thể tung chiêu để hạ gục người chơi.	
Actor	Monster	
Usecase liên quan	Tấn công Monster, Phun đạn	
Actor liên quan	Player	
Sự kiện kích hoạt	Trò chơi tại map có quái vật bắt đầu	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Nhấn bắt đầu trò chơi	1.1. Quái vật tấn công theo tầm đánh và

		sát thương được thiết lập
Kết quả	Quái vật tấn công nhân vật trong tầm đánh	

Bảng 16 - UC Tấn công Player

4.2. UC Tấn công Player – Phun đạn

ID	UC04.1	
Tên usecase	Phun đạn	
Mô tả	Quái vật phun đạn trong tầm đánh để hạ gục nhân vật	
Actor	Monster	
Usecase liên quan	Tấn công Player	
Sự kiện kích hoạt	Trò chơi tại map có quái vật bắt đầu	
Điều kiện tiên quyết	Người chơi nhấn bắt đầu chơi	
Luồng sự kiện	Tác nhân	Hệ thống
	1. Nhấn bắt đầu trò chơi	1.1. Quái vật được thiết lập phun đạn sẽ phun đạn trong tầm đánh được thiết lập 1.2. Đạn trúng nhân vật sẽ làm mất máu của nhân vật (trái tim)
Kết quả	Quái vật phun đạn tấn công nhân vật trong tầm đánh	

Bảng 17 - UC Phun đạn

CHƯƠNG III – HIỆN THỰC HỆ THỐNG

Bước 1: Xây dựng hệ thống.

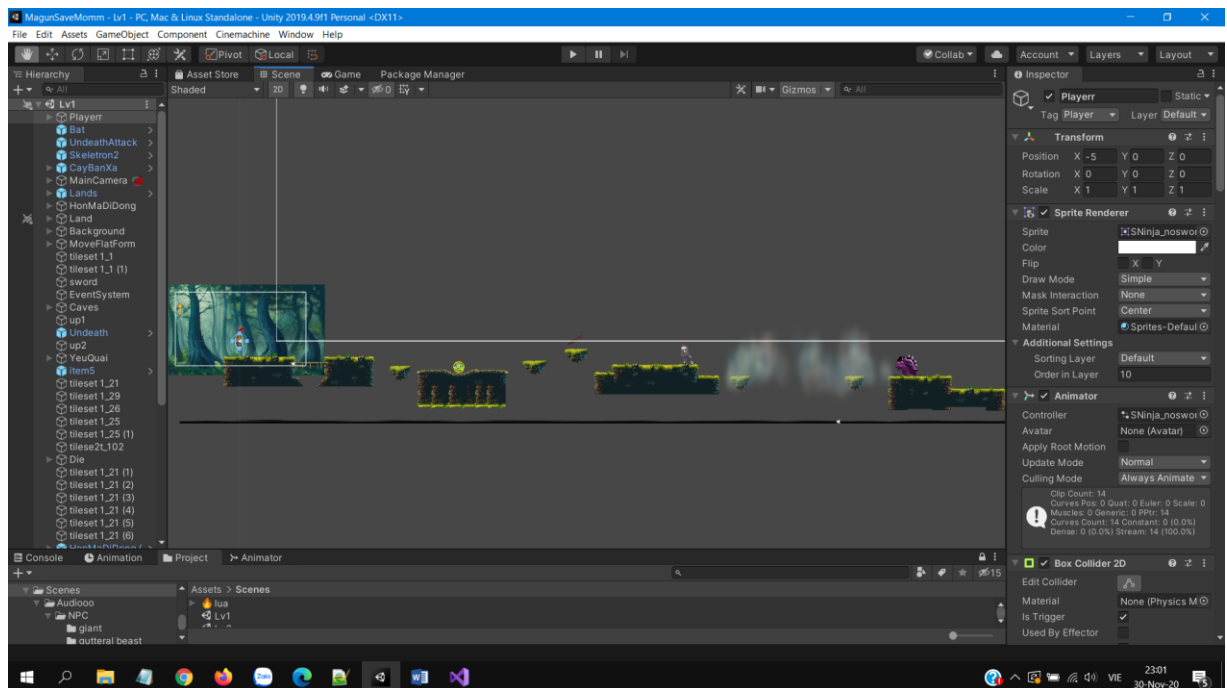
- + Phân tích trò chơi:
- + Thiết kế trò chơi:
- + Vẽ sơ đồ hệ thống.

Bước 2: Thu thập dữ liệu (assets, script, audio, ...).

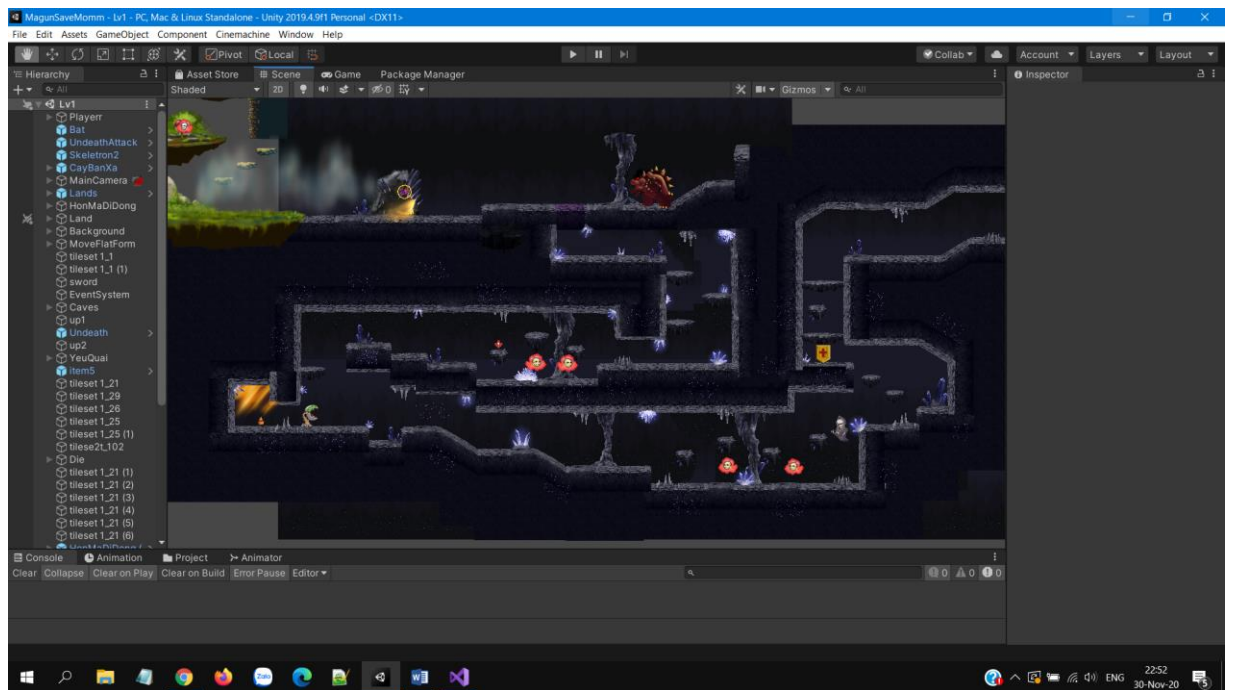
Dữ liệu chủ yếu sử dụng từ asset store của Unity.

Bước 3: Xây dựng UI.

- + Map level 1:

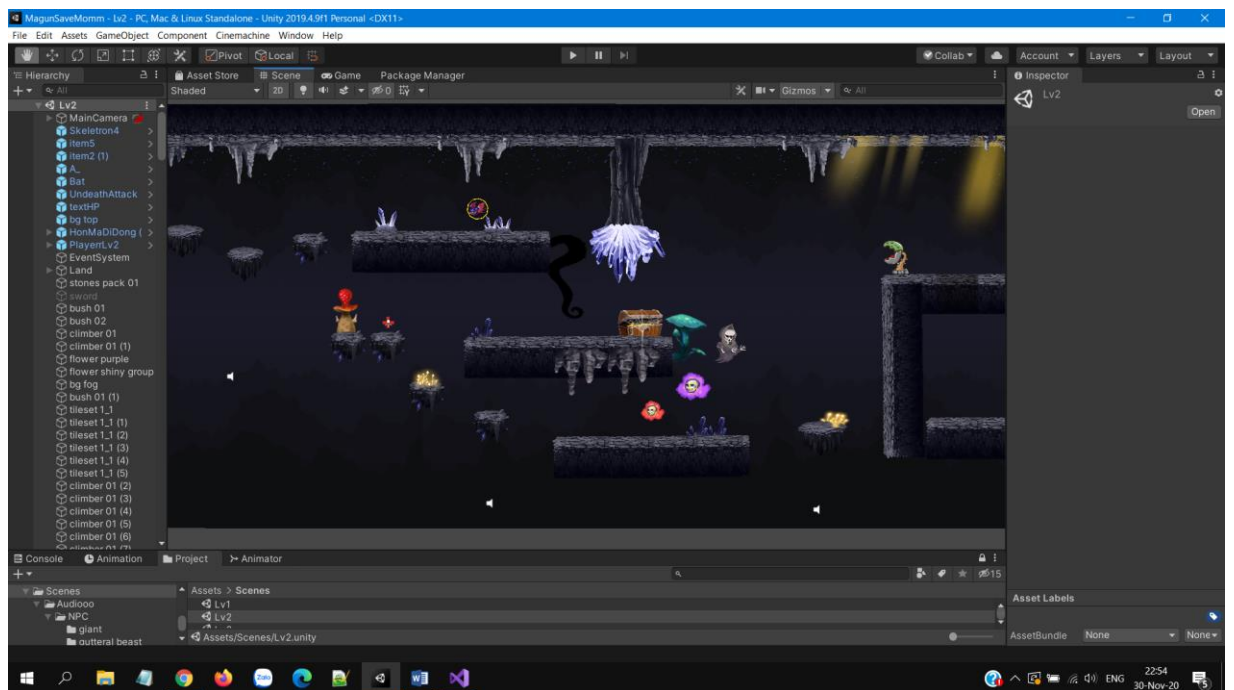


Hình 6 - Map Level 1-1

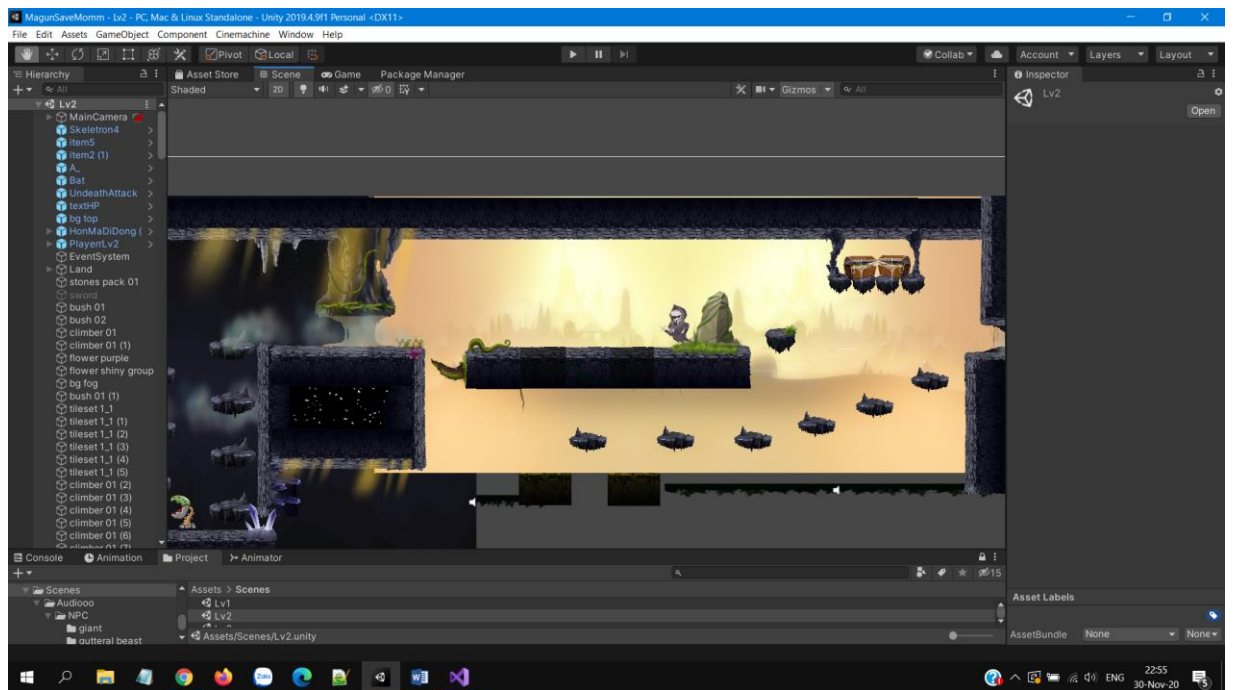


Hình 7 - Map Level 1-2

+ Map level 2:

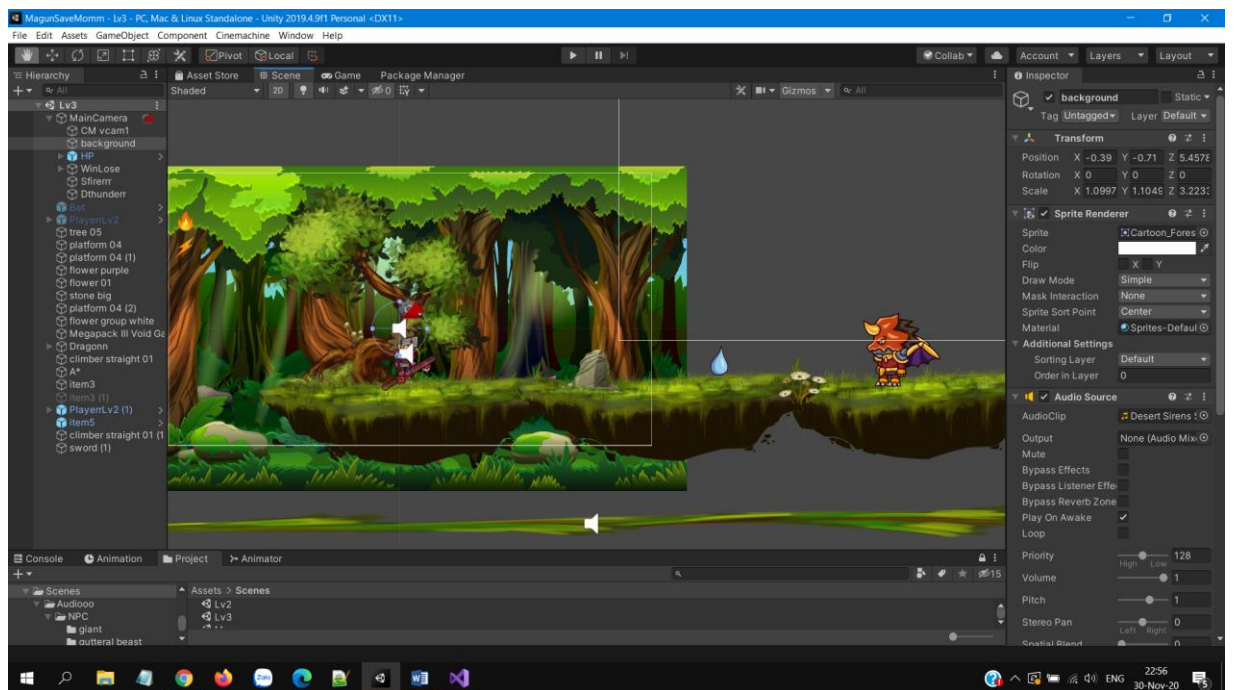


Hình 8 - Map Level 2-1



Hình 9 - Map level 2-2

+ Map level 3:



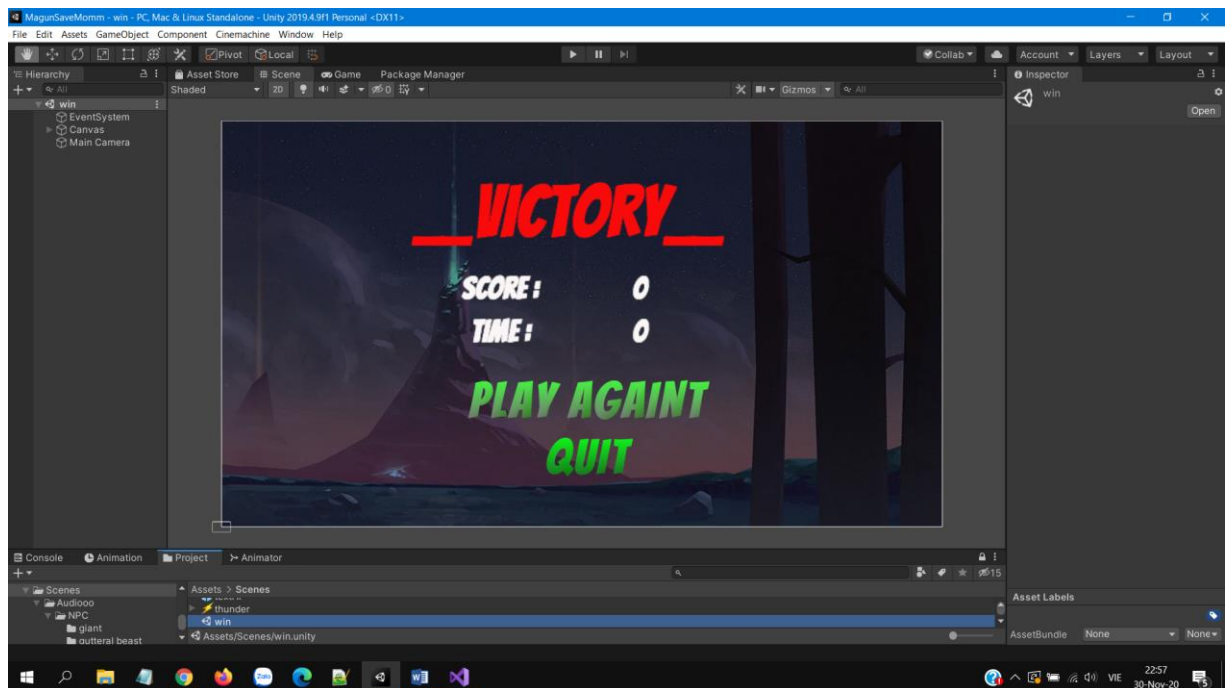
Hình 10 - Map level 3-1

+ Menu đầu game:



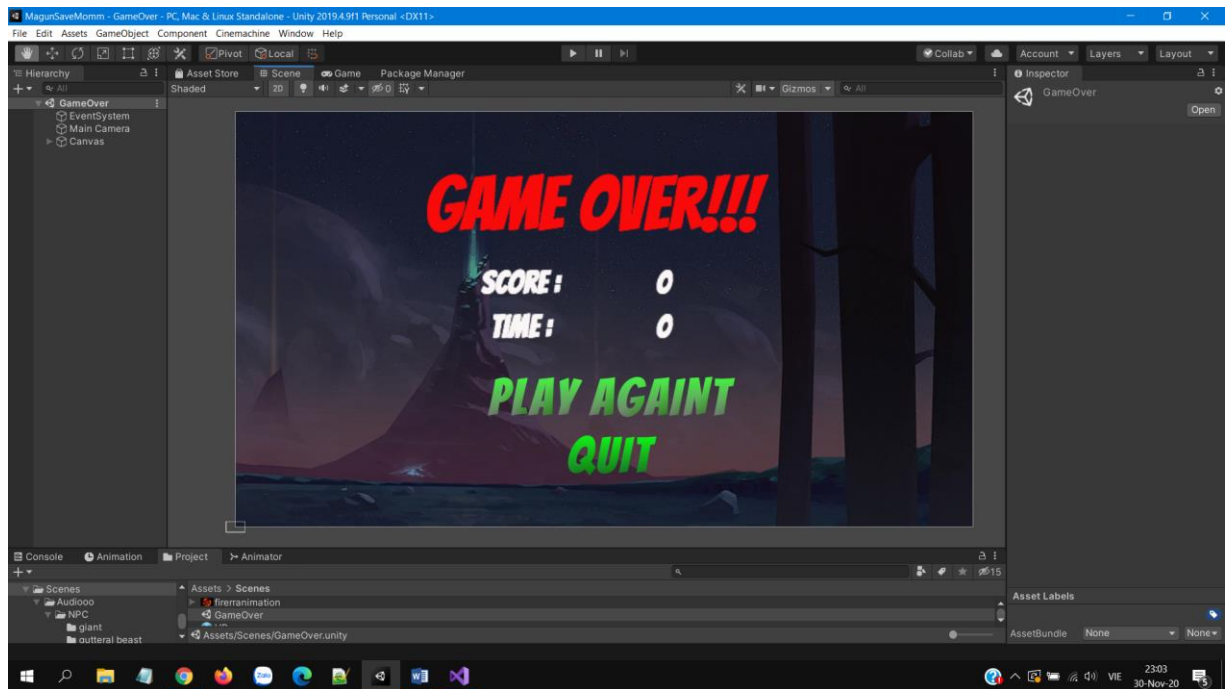
Hình 11 – Menu đầu tiên

+Menu End Game:



Hình 12 - Menu End game

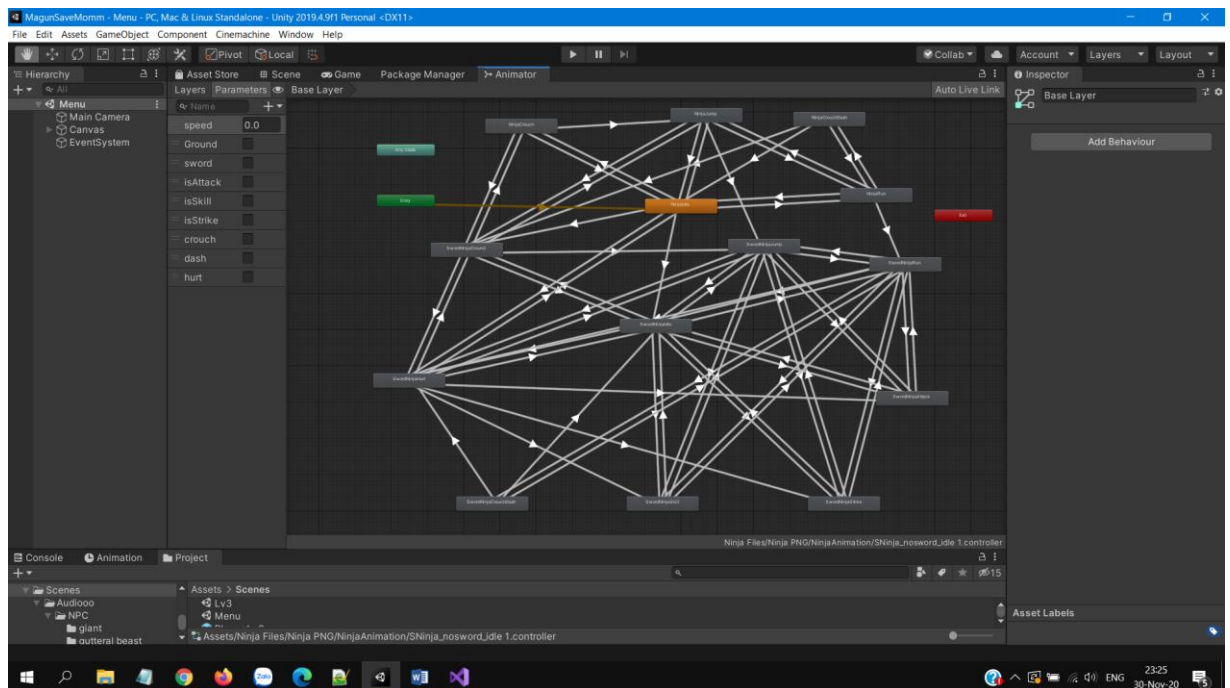
+ Menu Game Over:



Hình 13 - Menu Game Over

Bước 4: Viết code.

Mạch logic của animation nhân vật chính:



Hình 14 - Logic animation Player

Một số đoạn code chính:

- Nhân vật: các động tác di chuyển, tung chiêu với các phím mũi tên, A, S, D, L-Shift. Thiết lập các animation và các sự kiện liên quan.

```
using System.Collections;
using System.Collections.Generic;
using System.Runtime.Remoting.Messaging;
using System.Text;
using UnityEngine;

public class Mover : MonoBehaviour
{
    public float speed = 40f, maxspeed = 3, jumpPow = 300f;
    public bool grounded = true, faceright = true, doublejump = false;

    public GameObject playerr;
    public Rigidbody2D r2;
    public Animator anim;
    public GameObject boxAttack;

    public GameObject fire, fire2;
    public Transform fireSp, fireSp2;
    public float fireRate;
    private float timeRate;

    public float m, k;

    public CapsuleCollider2D capColli;
    public CapsuleCollider2D miniCaplli;

    public CountSkill countSkill;

    // Use this for initialization
    void Start()
    {
        r2 = gameObject.GetComponent<Rigidbody2D>();
        anim = gameObject.GetComponent<Animator>();
    }
    // Update is called once per frame
    void Update()
    {
        if (Input.GetKeyDown(KeyCode.R)) {
            playerr.transform.position = (new Vector2(-5, 0));
        }
        anim.SetBool("Ground", grounded);

        if (Input.GetKeyDown(KeyCode.UpArrow) && !anim.GetBool("dash"))
        {
            if (grounded)
            {
                grounded = false;
                doublejump = true;
                r2.AddForce(Vector2.up * jumpPow);
            }
            else
            {
                if (doublejump)
```

```

        {
            doublejump = false;
            r2.velocity = new Vector2(r2.velocity.x, 0);
            r2.AddForce(Vector2.up * jumpPow * 0.8f);
        }
    }

    if (Input.GetKey(KeyCode.DownArrow) && grounded)
    {
        anim.SetBool("crouch", true);
        anim.SetFloat("speed", 0);
        capColli.enabled = false;
        miniCaplli.enabled = true;
    }
    else {
        anim.SetBool("crouch", false);
        if (!anim.GetBool("dash"))
        {
            capColli.enabled = true;
            miniCaplli.enabled = false;
        }
    }

    if (Input.GetKeyDown(KeyCode.LeftShift) && grounded &&
        (Input.GetKey(KeyCode.RightArrow) || Input.GetKey(KeyCode.LeftArrow)))
    {
        anim.SetBool("dash", true);
        timeRate = Time.time;
    }
    if (anim.GetBool("dash"))
    {
        capColli.enabled = false;
        miniCaplli.enabled = true;
    }
    else {
        if (!(grounded))
        {
            capColli.enabled = true;
            miniCaplli.enabled = false;
        }
    }

    if (anim.GetBool("dash") && Time.time >= timeRate + 0.7f)
    {
        anim.SetBool("dash", false);
    }
}

[System.Obsolete]
void FixedUpdate()
{
    if (Input.GetKey(KeyCode.A) && !anim.GetBool("dash") &&
        !anim.GetBool("hurt"))
    {
        if (anim.GetBool("sword") == true)
        {
            anim.SetBool("isAttack", true);

```

```

        boxAttack.SetActive(true);
        timeRate = Time.time;
    }
}
else {
    anim.SetBool("isAttack", false);
    boxAttack.SetActive(false);
}
if (Time.time == timeRate + 0.3f)
{
    anim.SetBool("isAttack", false);
    boxAttack.SetActive(false);
}

if (Input.GetKey(KeyCode.S) && !anim.GetBool("dash") &&
countSkill.numOffFire>=1)
{
    anim.SetBool("isSkill", true);
    if (faceright && Time.time > timeRate && anim.GetBool("sword")==true)
    {
        timeRate = Time.time + fireRate;
        Instantiate(fire, fireSp.position, fireSp.rotation);
    }
    else
    {
        if (Time.time > timeRate && anim.GetBool("sword") == true)
        {
            timeRate = Time.time + fireRate;
            Instantiate(fire2, fireSp2.position, fireSp2.rotation);
        }
    }
}
else
{
    anim.SetBool("isSkill", false);
}

float vitri = playerr.transform.position.x;
if (Input.GetKeyDown(KeyCode.D) && anim.GetBool("sword") == true &&
!anim.GetBool("dash") && countSkill.countStrike==0)
{
    gameObject.GetComponent().Play();
    if (faceright)
    {
        m = 1;
    }
    else m = -1;
    anim.SetBool("isStrike", true);
    playerr.transform.position = new Vector2(vitri + m,
playerr.transform.position.y);
    boxAttack.SetActive(true);
    k = Time.time;
}

if (anim.GetBool("isStrike") && Time.time >= k + 0.3f)
{
    anim.SetBool("isStrike", false);
    boxAttack.SetActive(false);
}

```

```

float h = Input.GetAxis("Horizontal");

if (h!=0 && !anim.GetBool("crouch")) {
    anim.SetFloat("speed", 2);
    r2.transform.Translate(Vector2.right * h / 10f);
} else anim.SetFloat("speed", 0);

if (r2.velocity.x > maxspeed)
    r2.velocity = new Vector2(maxspeed, r2.velocity.y);
if (r2.velocity.x < -maxspeed)
    r2.velocity = new Vector2(-maxspeed, r2.velocity.y);

if (h > 0 && !faceright){
    Flip();
}
if (h < 0 && faceright){
    Flip();
}
}
public void Flip()
{
    faceright = !faceright;
    Vector3 Scale;
    Scale = transform.localScale;
    Scale.x *= -1;
    transform.localScale = Scale;
}
}

```

Ngoài ra còn nhiều Script cho quái, địa hình di chuyển, tấn công player, ghi điểm, tính thời gian,...

Bước 5: Test.

Tiến hành chơi map 1:

- + Di chuyển nhân vật, tung chiêu
- + Quái tấn công người chơi
- + Nhặt các vật phẩm hỗ trợ, (máu, lửa, vũ khí,...)
- + Tính điểm thời gian

Tiến hành chơi map 2

- + Di chuyển nhân vật, tung chiêu
- + Quái tấn công người chơi
- + Nhặt các vật phẩm hỗ trợ, (máu, lửa, vũ khí,...)
- + Tính điểm thời gian

Tiến hành chơi map cuối

- + Di chuyển nhân vật, tung chiêu

- + Quái tấn công người chơi

- + Nhặt các vật phẩm bổ trợ, (máu, lửa, vũ khí,...)

- + Tính điểm thời gian

Tiến hành kiểm tra menu options, menu game over, menu win game

Bước 6: Review.

Sửa các lỗi xảy ra trong quá trình test:

- + Collider

- + Thời gian, điểm số

- + Hướng tung chiêu, âm thanh

- + ...

Bước 7: Hoàn thành.

CHƯƠNG IV – KẾT QUẢ ĐẠT ĐƯỢC

Game MagunSaveMom (Ninja Cứu Mẹ) hoàn chỉnh đầy đủ level, âm thanh, hiệu ứng, không tồn tại các lỗi ảnh hưởng đến quá trình chơi game.



Hình 15 – Màn hình vào game

CHƯƠNG V – KẾT LUẬN

I. Ưu điểm:

Trò chơi dễ tiếp cận với mọi người chơi, mọi lứa tuổi với cách chơi đơn giản.

II. Nhược điểm:

Vì sử dụng các asset free nên UI vẫn còn chưa được bắt mắt, còn hạn chế trong việc sáng tạo.

III. Những gì đã làm được:

Ứng dụng những lý thuyết bổ ích cùng với những giờ luyện tập trên lớp thực hành vào trò chơi một cách thành công và hoàn thiện:

- + Animation cho nhân vật và quái.
- + Âm thanh trong trò chơi.
- + Tự thiết kế hình ảnh.
- + Sinh đối tượng.
- + AI cho quái.
- + ...

IV. Hướng phát triển cho tương lai:

Trong tương lai, trò chơi có thể phát triển trên đa nền tảng như PC, Android, IOS, Mac,... MagunSaveMom sẽ phát triển thêm nhiều map, nhiều chế độ chơi khác nhau cùng với các hiệu ứng đặc sắc giúp tăng trải nghiệm người chơi.

TỔNG KẾT

Game 2D offline có cốt truyện không còn là đề tài mới, tuy nhiên vẫn rất được người chơi đón nhận. Tựa game “Ninja cứu mẹ” của nhóm chúng em tuy còn một số khuyết điểm về đồ họa cũng như tiểu tiết, nhưng vẫn có thể làm nổi bật ưu điểm của một game 2D hấp dẫn: mang đến cảm giác cổ điển, cốt truyện cảm động, levels không dễ để vượt qua, nhân vật chính có nhiều chiêu thức...

Nhóm chúng em sẽ cố gắng cải thiện khuyết điểm và tiếp tục phát huy ưu điểm của “Ninja cứu mẹ”. Đồng thời, đây cũng là bước đầu để nhóm bước chân vào ngành công nghiệp game thú vị và khắc nghiệt.

TÀI LIỆU THAM KHẢO

1. <https://assetstore.unity.com/>
2. <https://www.codementor.io/unity-for-game-development-experts>
3. Slide bài giảng môn Phát triển trò chơi trường Đại học Tôn Đức Thắng.